Reprinted with permission from e-Connexion 2003 (2003.10.01)

Effective Processing of Compound Nouns during Japanese Morphological Analysis

Kazuo Aoki† , Hiroshi Inokawa††, Akihiro Nakayama†

†AP Globalization, Software Development Laboratory - Yamato (YSL)
††PvC Development, Software Development Laboratory - Yamato (YSL)

{Kazu Aoki, Hiroshi Inokawa, Akihiro Nakayama}/Japan/IBM@IBMJP

Abstract: NLP (Natural Language Processing) applications and components are necessary to implement different options in morphological analysis. One of these options is how compound words should be treated. For example, some users may want to retrieve long Japanese compound nouns such as “情報処理学会” ("Information Processing Society of Japan") as a single noun word, two noun words (“情報処理” + ”学会”: "Information Processing" + "Society"), or three noun words (“情報” + ”処理” + ”学会”: "Information" + "Processing" + "Society"). Usually, a keyword indexing search would prefer to return the smallest token sets, like “情報” + ”処理” + ”学会”, but an information extraction tool created for text mining may want to get an organization name like “情報処理学会”. Furthermore, user requirements vary on not only because of different usage priorities of NLP applications and components, but also on performance. We have developed an effective processing method of compound noun words in the Japanese lexical analysis engine found in LanguageWare V3, a worldwide linguistic component that supports both Indo-European and East Asian languages. Japanese processing is different from that used for Indo-European languages, because the target is to maximize performance specifically for Japanese compound noun words. Results have shown that this method can yield high performance: not breaking compound noun words, and breaking compound noun words take almost the same processing time. In this paper, we will report on the methodology and effectiveness of compound noun words processing in the Japanese morphological analysis.
Key words: Natural Language Processing, Compound words, Japanese morphological analysis

1. Introduction

Morphological analysis is widely used as a fundamental component for NLP (Natural Language Processing) applications in activities such as text mining, search, machine translation and Braille transcription. This is especially true for Chinese, Japanese and Thai, because they do not follow the Indo-European convention of entering spaces between words. Therefore, a word isolation method based on Indo-European writing conventions will not work. LanguageWare is IBM’s multilingual lexical and morphological analysis engine, and is now used in many NLP components and products. A cross-langugae compound words decomposition function is one of the optional functions of LanguageWare. Its purpose is to divide compound words into its smaller word constituents. For example, the German dictionary contains the compound word “Oberschulrat”,

which can be divided into “Ober”+”schul”+”rat”. The Japanese dictionary contains the compound word “情報処理学会”, which can be divided into “情報”+” 処理”+” 学会”.

Given Japanese compound noun words characteristics, when implementing this function for Japanese, we adopted a unique method unique that provides both high processing performance and easy dictionary maintenance.

2. Conventional Methods

Compound word processing is very important to Japanese, since compound noun words in Japanese are easily created and expanded. Methods to process these words have been researched and developed by several researchers in colleges and companies in Japan over a long period of time. Most of these methods can be grouped under one of two approaches. Both are performed after a morphological analysis (see [1],[2]). In one approach, a morphological analysis first breaks Japanese string to the smallest words by using the word dictionary without any compound noun words. Using the grammar dictionary, the compound words processing combines a subset of the tokens into one by using coincident static information. In the other approach, a morphological analysis first tokenizes a Japanese string into compound noun words by using the word dictionary that stores those noun words. Then the method divides it into smaller words by using coincident static information. In both cases, additional processing time is required and coincident static information must be prepared.

In Indo-European languages, decomposition is implemented at the end of lexical analysis in LanguageWare V3 (see [3]). Since the lexical analysis for Indo-European languages adopts a longest match method when selecting the best token, the decomposition method divides a compound word into some smaller words after selecting the best token. It uses morphotactics stored in a 4 bit sequence in the word dictionary.

3. Japanese Method in LanguageWare

Fig.1 shows a typical Japanese morphological analysis sequence, which is comprised of the following two stages:

1. Lookup word dictionary: this stage chooses all possible tokens found in the word dictionary, starting at the first character, and places them into a token-candidate list.
2. Select the best token and POS: this stage selects the most likely sentence with the best tokens and POS by calculating the connectivity cost between tokens using the minimum connectivity cost method (see [4]) to find a optimal path among all tokens in the token-candidate list.

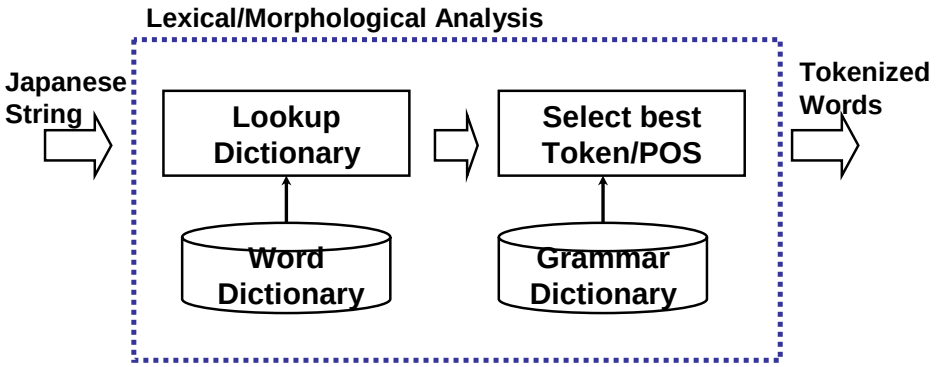

Fig.1 Overview of Morphological Analysis

3.1 Overview

Japanese lexical analysis in LanguageWare adopts the minimum connective cost method. This is well-known as a high quality method that uses a combination of a Dijkstra-like algorithm and Japanese grammatical rules. Using this method, we developed a simpler and more effective decomposition logic than that used for Indo-European languages.

Fig.2 shows an overview of this new approach. First, we added a dividable flag into the word dictionary, which consists of around 400,000 words. In order to reduce human workload, a semi-automatic program using the genetic algorithm was created (see [5]). Next, after preparing the word dictionary with a dividable gloss, we developed decomposition logic in the "lookup dictionary" stage. If the decomposition mode is ON when looking up the word dictionary, the logic checks each token that was found to see if its dividable flag is 1. If so, this token is not appended to the token-candidate list.

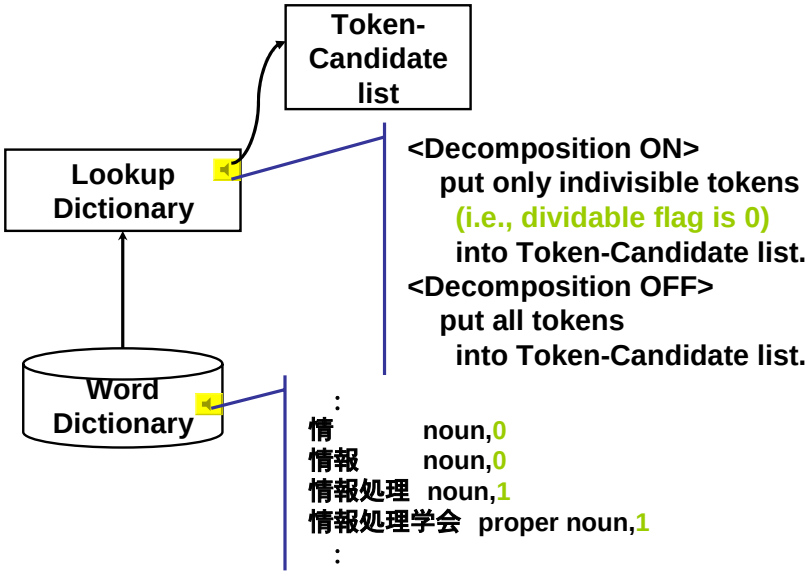

Fig.2 Overview of Japanese Method

3.2 Quality

When dividing compound noun words (Decomposition ON), only indivisible tokens (i.e., the flag value is 0) are placed in the token-candidate list. Consequently, there are no compound noun words in the token-candidate list. Thus only the smallest unit words available are placed in an optimal path and are selected correctly.

When not decompounding noun words (Decomposition OFF), all tokens found in the dictionary lookup are put into the token-candidate list. Consequently, there are both compound noun words as well as their unit tokens in the list. Only the compound noun words that are contained in an optimal path are correctly selected. This explains why compound words would be selected instead of unit words. Assuming that the minimum cost value of the optimal path to this word position was calculated and it is $g(x_i)$, and the connective costs between POSs are *"noun" + "noun" = a (> 0)* and *"noun" + "other POS" = b (> 0)*, total cost $f(x_i)$ concerning this word processing become

Case of compound noun words: $f1(x_i) = g(x_i)+b$
Case of unit noun words: $f2(x_i) = g(x_i)+a(+a+\ldots)+b$

Since $f1(x_i)$ is obviously less than $f2(x_i)$, compound noun words ($f1(x_i)$) is chosen.

Table.1 and Table.2 show the result of analyzing Japanese text on the Japanese lexical analysis in LanguageWare3.1, and Japanese compound noun words are correctly handled.

Table.1 Decomposition mode is OFF (default)

Surface form	POS#	JPOS#	Phrase
今年	3	5	1
から	6	39	0
人種差別	3	4	1
の	6	38	0
ない	4	23	1
十一	3	20	1
年間	3	29	0
の	6	38	0
無料	3	4	1
義務教育	3	4	0
を	6	39	0
始める	2	1	1
。	6	69	0

Table.2 Decomposition mode is ON

Surface form	POS#	JPOS#	Phrase
今年	3	5	1
から	6	39	0
人種	3	4	1
差別	3	4	0
の	6	38	0
ない	4	23	1
十一	3	20	1
年間	3	29	0
の	6	38	0
無料	3	4	1
義務	3	4	0
教育	3	4	0
を	6	39	0
始める	2	1	1
。	6	69	0

3.3 Performance

Experiments have been carried out to measure the performance of the Japanese decomposition method. We prepared 20 Japanese text files with differing lengths. The smallest text file is around 100K bytes (50K Japanese characters in UTF-16), which consists of Japanese sentences extracted from Japanese newspapers. By debugging Japanese lexical analysis, there are around 112,511 possible tokens in the token-candidate list, and about 6% of the tokens (6,734 tokens) have a dividable flag.

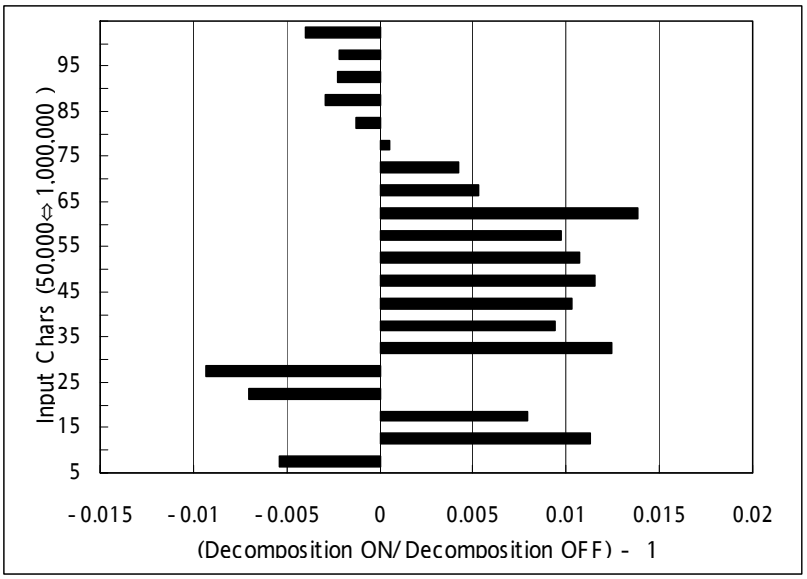

Fig.3 Performance Results

Fig.3 shows the performance and a comparison between decomposition OFF (default) and ON. The difference of the total processing time between OFF and ON is less than 1.5%., and that the logic for Japanese compound noun words is very efficient. The following are factors in the different processing time for our logic.

1. Increased time: the determination of the dividable flag value of all words
2. Reduced time: no connective cost calculation for dividable words

4. Conclusion

We have been able to develop the effective method of compound noun words in a Japanese lexical analysis in LanguageWare. Our experiments showed that our Japanese method can yield quite high performance and retain high accuracy rates.

References

[1] Hitachi Corp., Patent 1997-237277, Compound noun words processing method

[2] NTT Corp., Patent 2001-249921, Compound words processing method, device and record medium

[3] LanguageWare V3 User's Guide in 2003

[4] Hozumi Tanaka: NLP-elemental and practical, pp.2-15, The Institute of Electronics Information and Communication Engineers, 1999/3/25

[5] Tsuyoshi Matsuzaki: “Recognizing Japanese Compound Words using a Genetic Algorithm”, e-Connexion, 2003